

EGOS-DST:对话现象感知和模式引导的一步对话状态追踪算法

朱若尘, 杨长春, 张登辉

引用本文

朱若尘, 杨长春, 张登辉. EGOS-DST:对话现象感知和模式引导的一步对话状态追踪算法[J]. 计算机科学, 2022, 49(11A): 210900246-7.

ZHU Ruo-chen, YANG Chang-chun, ZHANG Deng-hui. EGOS-DST:Efficient Schema-guided Approach to One-step Dialogue State Tracking for Diverse Expressions [J]. Computer Science, 2022, 49(11A): 210900246-7.

相似文献推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[一种基于GPU的核苷酸分子系统发育树条件似然概率可扩展并行计算方法](#)

Scalable Parallel Computing Method for Conditional Likelihood Probability of Nucleotide Molecular Phylogenetic Tree Based on GPU

计算机科学, 2022, 49(11A): 210800189-7. <https://doi.org/10.11896/jsjcx.210800189>

[基于论据边界识别的立场分类研究](#)

Stance Detection Based on Argument Boundary Recognition

计算机科学, 2022, 49(11A): 210800180-5. <https://doi.org/10.11896/jsjcx.210800180>

[基于深度学习与文本计量的技术趋势分析](#)

Analysis of Technology Trends Based on Deep Learning and Text Measurement

计算机科学, 2022, 49(11A): 211100119-6. <https://doi.org/10.11896/jsjcx.211100119>

[一种融合多层次情感和主题信息的TS-AC-EWM在线商品排序方法](#)

TS-AC-EWM Online Product Ranking Method Based on Multi-level Emotion and Topic Information

计算机科学, 2022, 49(6A): 165-171. <https://doi.org/10.11896/jsjcx.210400238>

[融合Bert和图卷积的深度集成学习软件需求分类](#)

Deep Integrated Learning Software Requirement Classification Fusing Bert and Graph Convolution

计算机科学, 2022, 49(6A): 150-158. <https://doi.org/10.11896/jsjcx.210500065>

EGOS-DST: 对话现象感知和模式引导的一步对话状态追踪算法

朱若尘¹ 杨长春¹ 张登辉²

¹ 常州大学计算机与人工智能学院 江苏 常州 213159

² 浙江树人大学信息科技学院 杭州 310015

(vccorz@foxmail.com)

摘要 为了平衡过度依赖本体和完全舍弃本体两种极端方式,近期的对话状态追踪工作专注于混合方式。目前,这些混合方式忽略了一些特殊现象,比如值共享和推荐接受。此外,被广泛使用的槽位门机制使模型很难并行处理槽位,并且还会将误差传播到槽值生成步骤。针对以上问题,提出一种新的混合方式,它能够处理多样性表达、未知值、值共享和推荐接受4种不同对话现象。通过修改候选值集合和模型输入,模型不再依赖槽位门机制并且能够一步并行处理槽位。实验结果显示,模型在英文数据集 MultiWOZ 2.2 和 2.3 上分别达到了 57.7% 和 59.5% 的联合目标准确率,在中文数据集 RiSAWOZ 上达到了 68.1%,并且推理一次仅需 10ms。最后还分析了模型的鲁棒性,在 MultiWOZ 2.2 上的结果显示即使推荐错误率达到 15%,联合目标准确率仍有 55.4%。

关键词: 任务导向对话系统;对话状态追踪;BERT;并行计算;模式引导的对话

中图法分类号 TP391.1

EGOS-DST: Efficient Schema-guided Approach to One-step Dialogue State Tracking for Diverse Expressions

ZHU Ruo-chen¹, YANG Chang-chun¹ and ZHANG Deng-hui²

¹ School of Computer Science and Artificial Intelligence, Changzhou University, Changzhou, Jiangsu 213159, China

² College of Information Science and Technology, Zhejiang Shuren University, Hangzhou 310015, China

Abstract Recent dialogue state tracking works have focused on the hybrid approach to balance the two extreme methods(i. e., over-reliance on ontology and complete abandoning ontology). However, some special phenomena are ignored in these works. For instant, value sharing and recommendation acceptance. In addition, the widely used slot gate mechanism makes it difficult for the model to process slots in parallel and the mechanism also propagates errors to the slot value generation steps. This paper proposes a new hybrid approach that deals with four different phenomena, namely diverse value, unseen value, value sharing and recommendation acceptance. By modifying the candidate value set and the model input, our model can parallelly process slots in one step and no longer depend on the slot gate. Experimental results indicate that the model achieves 57.7% and 59.5% joint goal accuracy on the English dataset MultiWOZ 2.2 and 2.3, respectively, and reaches 68.1% on the Chinese dataset RiSAWOZ, with only 10ms infer time. Finally, the robustness of the model is analyzed. The results on MultiWOZ 2.2 show that the joint target accuracy rate is 55.4% when the recommendation error rate reaches 15%.

Keywords Task-oriented dialogue system, Dialogue state tracking, BERT, Parallel computing, Schema-guided dialogue

1 引言

虚拟助理能够帮助用户完成特定任务,包括但不限于天气查询、餐厅预订,它们通常支持大量服务且横跨多个领域。大型虚拟助理的兴起,例如 Siri, Cortana, 使构建可扩展的任务导向对话系统受到广泛关注^[1]。对话状态追踪(Dialogue State Tracking, DST)是任务导向对话系统的核心组件,它的目标是抽取并监控隐藏在对话历史中的用户目标,并将它们表示为对话状态,即一系列类似于(domain, slot, value)的三元组。对话状态被作为对话策略学习(Dialogue Policy Learning,

DPL)模块的输入,以此生成系统动作。系统动作决定自然语言生成(Natural Language Generation, NLG)模块所生成的回复,因此构建一个精确的 DST 对任务导向对话系统至关重要。

现存的 DST 方式根据是否使用本体(ontology)可以被划分为两类:基于本体(ontology-based)和无本体(ontology-free)。基于本体^[2-4]的方式预定义所有候选值,然后将 DST 简化为一个分类任务。然而,大量的可能值(比如日期、时间)使其很难应用于真实场景,并且这种方式还具有无法预测未知值(比如电影名称)的缺陷。为了解决这一问题,无本体的方式^[5-6]利用 Pointer Network (PtrNet)^[7]从对话上下文中

基金项目:浙江省公益技术研究计划(LGF21F020024);浙江树人大学青年学术团队项目

This work was supported by the Zhejiang Provincial Natural Science Foundation of China(LGF21F020024) and Youth Academic Team Project of Zhejiang Shuren University.

通信作者:张登辉(dhzhang@zjsru.edu.cn)

抽取槽值。这种方式不依赖本体,但它无法处理槽值未出现在对话中的现象,例如多样性表达(“I need *high priced* hotel.”)、值共享(“On the *same* day as the hotel stay.”)和推荐接受(“Yes, that sounds great!”)。上述两种方式具有各自的优势,但是它们走向了两种极端,即要么完全依赖本体,要么完全不使用,这显然与现实世界的服务有所冲突^[1]。与 Wang 等^[8]类似,本文强调 DST 模型应该合理地利用部分本体,而不是完全使用或者直接舍弃它。表 1 展示了上述 4 种不同现象,其中“r”和“t”代表“restaurant”和“taxi”,“...folk museum”指的是“cambridge and country folk museum”。

此外,无本体的 DST 方式高度依赖一个被称为槽位门(slot gate)的机制,其判断一个槽位是否在对话历史中被提到。如果被提到(即该机制的分类结果是 ptr),就利用 PtrNet 或其他方法生成槽值。近期有些工作^[9-10]已经展示了 ptr 的 F1 值在 95% 和 97% 之间。由于这种两步槽填充,无本体的 DST 方式正遭受着误差传播(即 slot gate→PtrNet)以及无法并行处理槽位的影响。

表 1 MultiWOZ 2.2 中一个包含 4 种对话现象的例子

Table 1 Example dialogue in MultiWOZ 2.2 with four dialogue phenomena

U: can you help me find a <i>pricy Singaporean</i> restaurant in Cambridge?	0 轮			
S: There are several. May I suggest <i>Curry Garden</i> is located in the Centre and has an expensive price range. Would you like a reservation?	7 轮			
U: Yes, can you book a table for 8 people on Saturday at 15:45?	8 轮			
S: A table is booked for Saturday for 15:45 at 8 and for 8 people. Reference number is B00KVHN.	9 轮			
U: I'm also looking for a place called <i>cambridge and country folk museum</i> .	10 轮			
S: Is there anything else I can help you with today?	13 轮			
U: I need a taxi to take me to <i>the restaurant</i> from <i>the museum</i> .	14 轮			
轮	槽位	用户表达	实际填充的值	现象
0	r-pricerange	pricy	expensive	多样性表达
0	r-food	Singaporean	Singaporean	未知值
8	r-name	Yes	Curry Garden	推荐接受
14	t-departure	the museum	...folk museum	值共享
14	t-destination	the restaurant	Curry Garden	值共享

为了处理上述问题,一方面,本文提出一种新的混合方式 EGOS-DST,利用基于本体和无本体的方式填充相应槽位,其中槽位是否可分类遵循数据集的模式(schema)^[11]。此外,本项工作还从前一轮的对话状态和推荐内存中拷贝槽值以分别处理值共享和推荐接受两种现象^[12-13]。值得注意的是,以往的工作假定系统(DPL)总是推荐准确的槽值对,这是不合理的。错误的推荐会引起另一种误差传播(即 DPL→NLG→PtrNet)。因此本文强调应该直接从推荐内存中获取槽值,而不是使用 PtrNet。另一方面,本文通过修改候选值集合和模型输入移除了槽位门,这切断了 slot gate→PtrNet 的误差传播路径。本文在广泛使用的 MultiWOZ 数据集和中文数据集 RiSAWOZ 上评估了模型的性能。综上所述,本文贡献如下:

(1) 提出了 EGOS-DST 模型,其能够感知 4 种对话现象,并且在多项实验数据集上取得了一流的表现;

(2) 通过引入功能性符号,EGOS-DST 能够一步并行处理槽位且不再依赖槽位门机制,大大提升了推理速度;

(3) 假定系统无法总是推荐准确的槽值对,本文展开了一系列实验以及分析了模型的鲁棒性;

(4) 验证了模型在中英文语料上构建 DST 的可行性。

2 相关工作

早期的 DST 研究常使用管道方式,将口语语音理解(Spoken Language Understanding, SLU)模块的输出作为 DST 的输入。Henderson 等^[14]认为这种方式会导致误差传播,即 SLU 阶段的误差会影响 DST。为了缓解这一问题,一系列的联合模型^[3,15-16]被提出。这些模型舍弃了 SLU 模块的输出,直接将对话历史作为 DST 的输入。虽然这些模型实现了性能的提高,但是它们都依赖一个预定义的本体。由于存在诸如无穷日期或周期性增加的音乐,因此很难应用这些模型。Xu 等^[5]首次将 Pointer Network(PtrNet)引入 DST 任务,它通过抽取对话中的字符串能够处理未知或不可枚举的值。随着一系列模型^[6,17]的提出,PtrNet 展现出了处理这些值的出色能力。

此后,一方面,一些工作^[18-20]开始使用混合的方式建模 DST。他们将所有槽位划分为可分类和不可分类两种,然后分别使用分类模型和 PtrNet 填充相应的槽位。但他们忽略了另外两种更含蓄的现象:值共享和推荐接受^[12-13]。

另一方面,以往的 DST 模型通常将整条对话历史作为输入,然后从头预测每轮的对话状态(dialogue-level DST)。虽然这保证了不同轮之间的对话状态不会互相干扰,但是冗长的对话历史带来了数据稀疏性的问题^[9],并且过长的输入序列导致模型需要花费大量时间执行推理步骤。有一系列工作^[9,21-22]试图使用前一轮的对话状态表示对话历史并且取得了具有竞争力的表现。同时,这些模型采用新的策略预测对话状态,它们将对话状态视为一个可以被选择性覆盖的内存,即使用当前轮生成的槽值对更新前一轮的对话状态(turn-level DST)。

此外,上述的大部分模型都需要使用一个被称为槽位门(slot gate)的机制,其决定一个槽位是否在对话历史中被提到。然而,数据不平衡的问题导致该机制的 F1 值在 95% 和 97% 之间^[9-10]。它会影响槽值生成步骤,类似于 SLU 模块的误差会影响 DST 的表现。通过对输入序列和候选值集合的修改,本文提出的模型可以完全不依赖槽位门。

最近提出的模型^[4,9,22-23]更多地关注槽位之间的相关性。虽然通过来自 Transformer 的 self-attention^[24]机制,本文提出的模型也有希望捕获输入序列中各个槽位之间的相关性,但这不是本项工作的关注重点。

3 方法设计

本文提出 EGOS-DST 模型,使用受到约束的本体^[1],称为模式(schema)。模式初步将所有槽位分为可分类和不可分类两种。然后本文进一步将任务导向对话归纳为 4 种现象,即多样性表达、未知值、值共享和推荐接受。多样性表达现象只出现在可分类槽位上;未知值现象只出现在不可分类槽位上;值共享现象和推荐接受现象可以出现在两种槽位上。

模型结构如图 1 所示,由 4 个模块组成,能够分别处理 4 种对应现象。得益于该结构,EGOS-DST 可以一步并行填充可分类槽位和不可分类槽位的槽值,不再依赖槽位门机制。EGOS-DST 的目标是在每轮生成一组槽值对,然后用它们更新前一轮的对话状态,更新机制详见第 3.7 节。

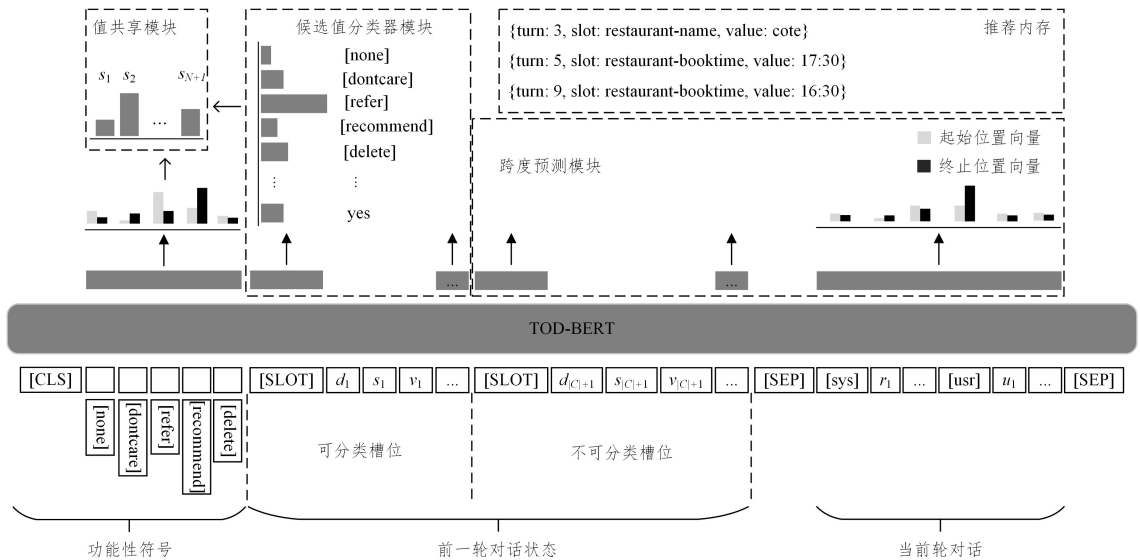


图1 EGOS-DST 的模型结构

Fig. 1 Architecture of EGOS-DST

3.1 形式化定义

3.1.1 功能性符号

为了阻止来自槽位门的误差,EGOS-DST 移除了它并引入了一些特殊值,称之为功能性符号。定义符号集合 $F = \{f_1, \dots, f_6\}$, 即 [CLS], [none], [dontcare], [refer], [recommend] 和 [delete], 并且将它们添加在候选值集合和模型输入中。[CLS]指的是BERT^[25]中的特殊符号。[none]和[dontcare]分别代表用户没有在语句中提到槽位和用户没有偏好。[refer]和[recommend]意味着模型需要从前一轮对话状态和推荐内存中分别处理值共享和推荐接受的现象。[delete]表示将槽值设置为[none]。

3.1.2 对话状态

本文将所有槽位定义为 $S = \{C, N\}$, 其中 C 代表可分类槽位集合, N 代表不可分类槽位集合。 t 轮时的对话状态被定义为 $\mathcal{D}_t = \{(d^i, s^i, v_i^s) \mid 1 \leq i \leq |S|\}$, 其中 d^i 和 s^i 分别代表第 i 个三元组的领域和槽位, v_i^s 表示 t 轮时 s^i 的槽值, $|S|$ 是槽位总量。注意, 对话状态中的三元组分为两部分。前者属于可分类, 后者属于不可分类。

3.1.3 当前轮对话

t 轮的对话被记为 $\mathcal{D}_t = [R, U_t]$, 其中 R 是系统回复, U_t 是用户语句, 这些语句分别以特殊符号 [sys] 和 [usr] 起始。

3.2 编码器

将前一轮对话状态记为 $B_{t-1} = B_{t-1}^1 \oplus \dots \oplus B_{t-1}^{|S|}$, 其中 $B_{t-1}^i = [\text{SLOT}] \oplus d^i \oplus s^i \oplus v_{t-1}^s$ 。 \oplus 是字符串拼接操作。 $v_{t-1}^s \in \{[\text{none}], [\text{dontcare}], [\text{value}]\}$, 其中 [value] 代表该槽位被填充了值。 [SLOT] 是一个特殊标记, 被用于聚合对应三元组的局部信息和上下文的全局信息。

本文使用 ToD-BERT^[26] 将 t 轮的输入序列编码为:

$$H_t = \text{ToDBERT}(F \oplus B_{t-1} \oplus [\text{SEP}] \oplus X_t \oplus [\text{SEP}]) \quad (1)$$

其中, ToD-BERT 是一个基于任务导向对话数据集的大型预训练模型。 [SEP] 分隔输入的不同部分。 $X_t = U_{t-1} \oplus R \oplus U_t$ 是 t 轮的对话上下文。 U_{t-1} 对一些特殊情况是有益的, 例如对于 “[sys] in what area? [usr] i would like the west please”, U_{t-1} 有可能提供 “area” 来自哪个领域的信息。

本文将每个 [SLOT] 的输出视为槽位表征, 那么可以得到可分类槽位表征 $H_t^C \in \mathbb{R}^{|C| \times d_{\text{hid}}}$ 和不可分类槽位表征 $H_t^N \in \mathbb{R}^{|N| \times d_{\text{hid}}}$ 。 $|C|$ 和 $|N|$ 分别是可分类槽位和不可分类槽位的数量。 d_{hid} 是隐藏状态的维度。

3.3 候选值分类器模块

候选值分类器模块被用于处理多样性表达现象。 假定每个可分类槽位 C 都有一个候选值集合 V^c 。 如果合并所有集合就能得到最终的候选值集合 V , 那么 t 轮时 C 的槽值预测就可以被视为分类任务, 其概率分布可以被并行计算为:

$$P_t^{\text{cat}} = \text{softmax}(H_t^C \cdot W_1 + b_1) \in \mathbb{R}^{|C| \times |V|} \quad (2)$$

由于 V 是固定的, 因此该模块能够生成一个唯一值。 例如, 对于语义相同的表达 “high end” 和 “pricy”, 预测结果只能是 “expensive”。 最后优化交叉熵损失, 被记为 \mathcal{L}_{cat} 。

3.4 跨度预测模块

跨度预测模块利用 PtrNet 从对话中抽取未知值。 该模块将 H_t 作为输入, 然后与 H_t^N 交互, 以此分别得到起始位置和终止位置。 跨度 (span) 内的字符串就是预测出的槽值。 在 t 轮时, 同样可以并行计算不可分类槽位 N 的跨度:

$$\begin{cases} [Q_t^1, Q_t^2] = \text{FFN}(H_t^N) \in \mathbb{R}^{|N| \times 2d_{\text{hid}}} \\ P_t^{\text{start}} = \text{softmax}(Q_t^1 \cdot H_t^T) \in \mathbb{R}^{|N| \times |X_t|} \\ P_t^{\text{end}} = \text{softmax}(Q_t^2 \cdot H_t^T) \in \mathbb{R}^{|N| \times |X_t|} \\ \text{start}_t = \arg \max(P_t^{\text{start}}) \in \mathbb{R}^{|N|} \\ \text{end}_t = \arg \max(P_t^{\text{end}}) \in \mathbb{R}^{|N|} \end{cases} \quad (3)$$

其中, $\text{FFN}(x) = \text{GELU}(x \cdot W_2 + b_2) \cdot W_3 + b_3$ 。 值得注意的是, 本文将 [CLS] 和前一轮对话状态序列位置上的分布置为一 $-\infty$, 这是因为它们不可能是槽值的一部分。 最后, 优化交叉熵损失, 被记为 $\mathcal{L}_{\text{noncat}} = \mathcal{L}_{\text{start}} + \mathcal{L}_{\text{end}}$ 。

3.5 值共享模块

如表 1 所列, 还存在更复杂的对话。 在出现值共享现象时, 两个槽位的槽值是相等的。 值共享模块旨在预测可能的槽位, 以此建立两个槽位之间的关联。 该模块的处理过程见图 2。 如果候选值分类器模块或者跨度预测模块的预测结果是 [refer], 那么槽位 S^k 在所有可能槽位上的概率分布可以表示为:

$$p_{i,k}^{\text{refer}} = \text{softmax}(H_i^k \cdot W_4 + b_4) \in \mathbb{R}^{N+1} \quad (4)$$

其中,1 指的是[none]。如果 $S^n = \arg \max(p_{i,k}^{\text{refer}})$,那么 S^k 的槽值就直接从 S^n 复制。类似地,可以得到交叉熵损失 $\mathcal{L}_{\text{refer}}$ 。

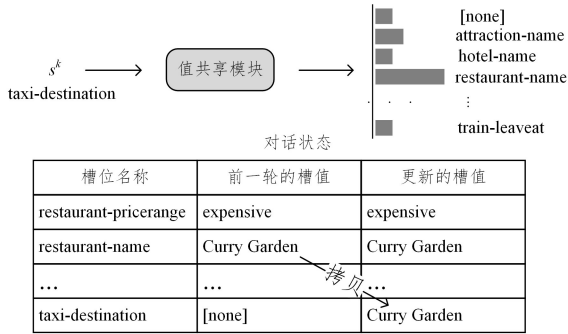


图2 值共享模块的处理过程

Fig. 2 Process of value sharing module

注意,由于在 MultiWOZ 2.2 中,可分类槽位没有“copy_from”标签,我们使用了一个简单的脚本生成该标签(如果用户在语句中提到了可分类槽位并且其中出现单词“same”,就生成“copy_from”标签)。“copy_from”的值是另一个槽位,这代表两个槽位共享槽值。

3.6 推荐内存

推荐内存 $\mathcal{M}_t = \{(S^m, v_t^m) \mid (1 \leq m \leq M)\}$ 中存储了 t 轮之前所有的系统推荐,其中 M 是推荐次数。如果候选值分类器模块或者跨度预测模块的预测结果为[recommend],那么就使用槽位 S^m 逆序匹配 \mathcal{M}_t 中的槽位。如果槽位名称相等, S^m 就填充对应槽位的值。如果没有槽位被匹配,就填充[none]。

3.7 对话状态更新

在每轮 t ,模型会为每个槽位 s 预测一个值 $v_{t,s}^{\text{temp}}$ (即除[CLS]外的功能性符号或某个字符串),以此更新新一轮的对话状态。那么 s 的槽值 v_t^s 可以按照以下规则更新或者保持不变:

$$v_t^s = \begin{cases} v_{t-1}^s, & \text{if } v_{t,s}^{\text{temp}} = [\text{none}] \\ [\text{dontcare}], & \text{if } v_{t,s}^{\text{temp}} = [\text{dontcare}] \\ \text{from DS}, & \text{if } v_{t,s}^{\text{temp}} = [\text{refer}] \\ \text{from RM}, & \text{if } v_{t,s}^{\text{temp}} = [\text{recommend}] \\ [\text{none}], & \text{if } v_{t,s}^{\text{temp}} = [\text{delete}] \\ v_{t,s}^{\text{temp}}, & \text{otherwise} \end{cases} \quad (5)$$

其中,“from DS”和“from RM”分别表示从对话状态和推荐内存中获取槽值。

4 实验

4.1 数据集

本文在 MultiWOZ 2.2^[11], MultiWOZ 2.3^[10] 和 RiSAWOZ^[27] 上评估提出的模型。MultiWOZ 包含超过 10 000 条被标注的对话并且横跨 8 个领域。此外,MultiWOZ 2.2 移除了本体中不可分类槽位的候选值集合,这种受约束的本体被称为模式(schema)。MultiWOZ 2.3 未修改本体。我们使用 MultiWOZ 2.2 中的规则生成了模式。该规则将候选值数量小于 50 的槽位视为可分类槽位,其他的视为不可分类槽位。遵循 Wu 等^[17]的工作,本文仅使用 5 个领域(hotel, train, restaurant, attraction, taxi),这是因为其余领域只具有少量对话。RiSAWOZ 是一个中文任务导向对话数据集,包含 1 1200 条

被标注的人人对话,超过 150 000 条语句且横跨 12 个领域。本文使用类似的规则生成了模式,除了将阈值 50 改为 30。

4.2 训练

由于受到计算设备内存(12 GB)的限制,本项工作使用 ToD-BERT^[26] 的蒸馏版本作为编码器。该预训练模型有 6 层隐藏层以及 12 个自注意力头,每层拥有 768 个单元。梯度截断的最大范数设置为 5.0。与以往工作类似,也对输入序列(除功能性符号和前一轮对话状态外)使用 word dropout 技术^[28],概率设置为 0.1。批次大小和 epoch 分别设置为 16 和 3。使用 Adam 优化器^[29]和斜三角学习率^[30]优化联合损失 $\mathcal{L} = \mathcal{L}_{\text{cat}} + \mathcal{L}_{\text{noncat}} + \mathcal{L}_{\text{refer}}$,最大学习率设置为 8×10^{-5} 。实验结果可以在 1h 内使用一个 12GB P100 GPU 复现。

训练时,将真实的前一轮对话状态输入模型。推理时,使用预测出的前一轮对话状态。

4.3 主要结果

本文使用联合目标准确率(Joint Goal Accuracy, JGA)指标评估模型表现。在每轮,当且仅当所有(domain, slot, value)三元组被准确地预测出时, JGA 才为 1, 否则为 0。结果(运行 10 次取均值)列于表 2、表 3 中,被 * 标记的模型属于无本体方式。

表 2 MultiWOZ 2.2 和 2.3 测试集上的 JGA 对比

Table 2 JGA comparison on MultiWOZ 2.2 and 2.3 test sets (单位: %)

Model	MWOZ 2.2	MWOZ 2.3
SGD-baseline ^[19]	42.0	—
TRADE * ^[17]	45.4	49.2
COMER * ^[31]	—	50.2
DSTQA ^[32]	—	51.8
SUMBT ^[33]	49.7	52.9
DS-DST ^[20]	51.7	—
Seq2Seq-DU ^[34]	54.4	—
STAR ^[4]	55.4	—
SOM-DST ^[21]	—	55.5
SAVN ^[8]	—	58.0
TripPy * ^[13]	—	63.0
EGOS-DST(SG)	57.3	59.3
EGOS-DST	57.7	59.5

表 3 RiSAWOZ 测试集上的 JGA 对比

Table 3 JGA comparison on RiSAWOZ test set

Model	RiSAWOZ
TRADE(rand init) * ^[17]	58.2
TRADE(fastText) * ^[17]	60.1
MLCSG(fastText) * ^[35]	66.2
EGOS-DST	68.1

表 2 显示在 MultiWOZ 2.2 上 EGOS-DST 达到了最佳的表现,并且与一流模型 STAR 相比提升了 2.3%。在 MultiWOZ 2.3 上, EGOS-DST 的表现仅低于 TripPy。然而, TripPy 完全依赖拷贝机制,属于无本体方式,其无法预测用户没有表达在语句中的槽值,因此它需要利用一个标签字典(label map)去纠正预测^[4](例如将“high end”映射为“expensive”)。实际上很难枚举诸如“high end”的槽值变体,这使模型无法应用于真实场景。表 3 显示 EGOS-DST 相比于 MLCSG 提升了 1.9%。

此外,本文还对比了 Cat-JGA 和 Noncat-JGA 两个指标,它们分别计算在可分类槽位和不可分类槽位上的 JGA。表 4 显示 EGOS-DST 没有在 Cat-JGA 上达到最佳的表现,这可能是因为本文方法将可分类槽位的填充简单地视为一个分类

任务。在 Noncat-JGA 指标上,EGOS-DST 显著地超过了其他模型,这得益于它能够处理值共享和推荐接受两种现象(这两种现象大量地出现在不可分类槽位上)。其中后者占主要贡献,这是因为:1)推荐接受现象的出现次数远大于值共享;2)与以往的 DST 方式相同,本文假设系统总是能够生成准确的槽值对。本文将在第 5.2 节进一步讨论系统推荐的问题。

表 4 MultiWOZ 2.2 测试集上的 Cat-JGA 和 Noncat-JGA

Table 4 Cat-JGA and Noncat-JGA of various models on

MultiWOZ 2.2 test set

(单位:%)

Model	Cat-JGA	Noncat-JGA
SGD-baseline	57.0	66.1
TRADE	62.8	66.6
DS-DST	70.6	70.1
Seq2Seq-DU	75.8	71.1
EGOS-DST	69.1	80.5

5 分析

5.1 移除槽位门的影响

以往大部分 DST 模型高度依赖一个被称为槽位门的机制。本节展示每个槽位的准确率(per slot accuracy)并分析移除槽位门的影响。如表 5 所列,在 EGOS-DST 上大部分槽位的准确率高于 EGOS-DST(SG)。其中,EGOS-DST(SG)表示应用了槽位门,“r”指的是“restaurant”。表的上半部分包含可分类槽位,下半部分包含不可分类槽位,最后一行指平均准确率。此外,表 2 也说明了 EGOS-DST 的表现好于 EGOS-DST(SG)。综上所述,移除槽位门确实能够改善模型的表现。

表 5 不同模型在 MultiWOZ 2.2 上每个槽位的准确率

Table 5 Per slot accuracy on Multiwoz 2.2 test set

(单位:%)

Slot name	TRADE	EGOS(SG)	EGOS
attraction-area	96.91	96.49	96.73
attraction-type	95.89	96.80	96.83
hotel-area	95.15	95.91	96.16
hotel-bookday	98.91	99.14	99.17
hotel-bookpeople	98.60	99.16	99.17
hotel-bookstay	99.33	99.39	99.43
hotel-internet	96.24	96.08	96.28
hotel-parking	96.31	96.87	97.09
hotel-pricerange	96.15	96.94	96.53
hotel-stars	97.37	98.13	98.09
hotel-type	93.06	94.54	94.10
r-area	96.65	97.17	96.98
r-bookday	99.23	99.40	99.21
r-bookpeople	99.16	99.05	99.24
r-pricerange	97.10	97.47	97.81
train-bookpeople	98.10	97.13	97.58
train-day	98.75	98.92	99.02
train-departure	97.83	98.22	98.47
train-destination	98.64	98.56	98.64
attraction-name	91.88	97.75	97.25
hotel-name	94.00	97.75	97.78
r-booktime	98.85	99.36	99.18
r-food	97.03	97.25	97.38
r-name	91.34	97.03	97.09
taxi-arriveby	99.20	99.13	99.29
taxi-departure	96.81	97.59	97.60
taxi-destination	97.05	97.81	97.98
taxi-leaveat	98.85	99.29	99.37
train-arriveby	96.08	96.77	96.81
train-leaveat	93.69	94.12	94.24
slot accuracy	96.81	97.64	97.68

以往的模型在预测槽值之前,需要先执行槽位门机制。该机制使模型很难并行处理所有槽位,特别对于类似 TripPy 这种需要使用多种机制的模型。EGOS-DST 通过引入功能性符号移除了槽位门,它可以并行地填充槽位。本文将在第 5.3 节展示 EGOS-DST 的推理速度。

5.2 系统推荐误差的影响

在真实的场景中,系统会给用户一些推荐供其选择。据统计,attraction,hotel 以及 restaurant 的名称占绝大多数。如表 5 所列,在这 3 个槽位的准确率上,EGOS-DST 相比于 TRADE 获得了极大的提升。然而系统无法总是生成准确的槽值对。因此,本文执行一系列的实验来评估模型的鲁棒性。首先从训练集中收集系统可能会推荐的槽位 S^{rec} ,然后在 DST 预测每轮的对话状态之前,以 θ 的概率将真实(ground truth)的推荐(即由系统在上一轮生成的槽值对)替换为一个随机槽值对,其中 θ 是推荐错误率。随机槽值对中的槽位采样自 S^{rec} ,槽值是任意的占位符。

实验结果展示在表 6 中,其中 0%代表使用真实推荐。“-rec”代表使用 PtrNet 处理推荐接受,而不是从推荐内存拷贝。结果基于 MultiWOZ 2.2。令人意外的是,即使系统具有 15%的推荐错误率,模型依旧表现良好。本文认为与以下两方面有关:1)训练集中的标签会指导模型判断哪些槽位是 [recommend] 类型,这不会受到系统推荐的影响;2)只有 3 个槽位(即 attraction-name,hotel-name 和 restaurant-name)受到推荐的严重影响。

表 6 EGOS-DST 在不同推荐错误率下的 JGA

Table 6 JGA of model with different recommendation error rates

(单位:%)

推荐错误率	0	1	2	5	10	15	20	-rec
JGA	57.74	57.53	57.31	57.00	55.88	55.39	54.34	52.07

5.3 推理速度分析

表 7 列出了 EGOS-DST 和其他模型的推理速度,所有结果衡量于 Tesla P100,批次大小为 1。TRADE 是一个经典模型,被视为基准;SOM-DST 是近期最快的基于 BERT 的模型;STAR 是目前表现最佳的基于本体的模型。小括号中的数字是每轮对话中 EGOS-DST 的运行时间。其余时间被用于处理槽值,例如将单词 id 转换为单词。

表 7 MultiWOZ 2.2 上每轮对话的推理时间和模型大小

Table 7 Inference time and model size per dialogue turn on

MultiWOZ 2.2 test set

Model	Size/MB	Latency/ms
TRADE	10	274.0
SOM-DST	113	22.0
STAR	146	25.0
EGOS-DST(SG)	74	10.7(10.5)
EGOS-DST	74	10.4(8.4)

TRADE^[17]的推理时间是 EGOS-DST 的 26.3 倍,这得益于 EGOS-DST 能够一步并行处理槽位及其只接收当前轮对话和前一轮用户语句作为输入。由于 STAR^[4]是基于本体的模型,因此它无需槽位门并且能够并行处理槽位。然而,为了捕获槽位之间的相关性,STAR 在 BERT 之上还堆叠了多层自注意力层,导致其推理时间接近于 SOM-DST^[21]。EGOS-DST 快于 SOM-DST 2.1 倍,这得益于 EGOS-DST 的一步 DST 方式以及 DistilBERT^[36]的推理速度。

结束语 本文提出了一种新的混合方式 EGOS-DST, 其不仅能够处理 4 种不同的现象, 而且还不依赖槽位门机制。主要实验结果表明: 1) EGOS-DST 达到了一流的表现; 2) 槽位门的移除可以大幅提升模型的推理速度且可以进一步提升联合目标准确率。本文还分析了系统推荐误差对模型的影响。实验结果显示 EGOS-DST 具有不错的鲁棒性。最后, 本文也验证了 EGOS-DST 在中英文语料上构建 DST 的可行性。下一步将改进候选值分类器模块以加强模型处理可分类槽位的能力, 还会引入模式描述和槽位关联以增强槽位的表征能力。

参考文献

- [1] RASTOGI A, ZANG X, SUNKARA S, et al. Schema-guided dialogue state tracking task at DSTC8 [J]. arXiv: 2002. 01359, 2020.
- [2] ZHONG V, XIONG C, SOCHER R. Global-Locally Self-Attentive Encoder for Dialogue State Tracking [C] // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne: Association for Computational Linguistics, 2018: 1458-1467.
- [3] SHAN Y, LI Z, ZHANG J, et al. A Contextual Hierarchical Attention Network with Adaptive Objective for Dialogue State Tracking [C] // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, 2020: 6322-6333.
- [4] YE F, MANOTUMRUKSA J, ZHANG Q, et al. Slot Self-Attentive Dialogue State Tracking [J]. arXiv: 2101. 09374, 2021.
- [5] XU P, HU Q. An End-to-end Approach for Handling Unknown Slot Values in Dialogue State Tracking [C] // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne: Association for Computational Linguistics, 2018: 1448-1457.
- [6] CHAO G L, LANE I. Bert-dst: Scalable end-to-end dialogue state tracking with bidirectional encoder representations from transformer [J]. arXiv: 1907. 03040, 2019.
- [7] VINYALS O, FORTUNATO M, JAITLEY N. Pointer Networks [C] // Advances in Neural Information Processing Systems. Montréal: Curran Associates, Inc., 2015: 2692-2700.
- [8] WANG Y, GUO Y, ZHU S. Slot Attention with Value Normalization for Multi-Domain Dialogue State Tracking [C] // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Online: Association for Computational Linguistics, 2020: 3019-3028.
- [9] ZHU S, LI J, CHEN L, et al. Efficient Context and Schema Fusion Networks for Multi-Domain Dialogue State Tracking [C] // Findings of the Association for Computational Linguistics: EMNLP 2020. Online: Association for Computational Linguistics, 2020: 766-781.
- [10] HAN T, LIU X, TAKANOBU R, et al. MultiWOZ 2.3: A multi-domain task-oriented dataset enhanced with annotation corrections and co-reference annotation [J]. arXiv: 2010. 05594, 2020.
- [11] ZANG X, RASTOGI A, SUNKARA S, et al. MultiWOZ 2.2: A Dialogue Dataset with Additional Annotation Corrections and State Tracking Baselines [C] // Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI. Online: Association for Computational Linguistics, 2020: 109-117.
- [12] OUYANG Y, CHEN M, DAI X, et al. Dialogue State Tracking with Explicit Slot Connection Modeling [C] // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, 2020: 34-40.
- [13] HECK M, NIEKERK C, LUBIS N, et al. TripPy: A Triple Copy Strategy for Value Independent Neural Dialog State Tracking [C] // Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue. 1st virtual meeting: Association for Computational Linguistics, 2020: 35-44.
- [14] HENDERSON M, THOMSON B, YOUNG S. Word-Based Dialogue State Tracking with Recurrent Neural Networks [C] // Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL). Philadelphia, PA: Association for Computational Linguistics, 2014: 292-299.
- [15] ZILKA L, JURCICEK F. Incremental LSTM-based dialog state tracker [C] // 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). Scottsdale, AZ: IEEE Press, 2015: 757-762.
- [16] MRKŠIĆ N, Ó SÉAGHDHA D, WEN T H, et al. Neural Belief Tracker: Data-Driven Dialogue State Tracking [C] // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver: Association for Computational Linguistics, 2017: 1777-1788.
- [17] WU C-S, MADOTTO A, HOSSEINI-ASL E, et al. Transferable Multi-Domain State Generator for Task-Oriented Dialogue Systems [C] // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019: 808-819.
- [18] GOEL R, PAUL S, HAKKANI-TÜR D. Hyst: A hybrid approach for flexible and accurate dialogue state tracking [J]. arXiv: 1907. 00883, 2019.
- [19] RASTOGI A, ZANG X, SUNKARA S, et al. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset [C] // Proceedings of the AAAI Conference on Artificial Intelligence. New York: AAAI Press, 2020: 8689-8696.
- [20] ZHANG J, HASHIMOTO K, WU C S, et al. Find or Classify? Dual Strategy for Slot-Value Predictions on Multi-Domain Dialogue State Tracking [C] // Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics. Barcelona, Spain (Online): Association for Computational Linguistics, 2020: 154-167.
- [21] KIM S, YANG S, KIM G, et al. Efficient Dialogue State Tracking by Selectively Overwriting Memory [C] // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, 2020: 567-582.
- [22] ZENG Y, NIE J Y. Multi-domain dialogue state tracking based on state graph [J]. arXiv: 2010. 11137, 2020.
- [23] HU J, YANG Y, CHEN C, et al. SAS: Dialogue State Tracking via Slot Attention and Slot Information Sharing [C] // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, 2020: 6366-6375.

- [24] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is All you Need[C]// Advances in Neural Information Processing Systems. Long Beach, CA: Curran Associates, Inc., 2017; 5998-6008.
- [25] DEVLIN J, CHANG M W, LEE K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics, 2019; 4171-4186.
- [26] WU C S, HOI S C H, SOCHER R, et al. TOD-BERT: Pre-trained Natural Language Understanding for Task-Oriented Dialogue [C]// Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Online: Association for Computational Linguistics, 2020; 917-929.
- [27] QUAN J, ZHANG S, CAO Q, et al. RISA WOZ: A Large-Scale Multi-Domain Wizard-of-Oz Dataset with Rich Semantic Annotations for Task-Oriented Dialogue Modeling [C]// Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Online: Association for Computational Linguistics, 2020; 930-940.
- [28] BOWMAN S R, VILNIS L, VINYALS O, et al. Generating Sentences from a Continuous Space [C]// Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning. Berlin: Association for Computational Linguistics, 2016; 10-21.
- [29] KINGMA D P, BA J. Adam: A method for stochastic optimization [J]. arXiv:1412.6980, 2014.
- [30] HOWARD J, RUDER S. Universal Language Model Fine-tuning for Text Classification [C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne: Association for Computational Linguistics, 2018; 328-339.
- [31] REN L, NI J, MCAULEY J. Scalable and Accurate Dialogue State Tracking via Hierarchical Sequence Generation [C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong: Association for Computational Linguistics, 2019; 1876-1885.
- [32] ZHOU L, SMALL K. Multi-domain dialogue state tracking as dynamic knowledge graph enhanced question answering [J]. arXiv:1911.06192, 2019.
- [33] LEE H, LEE J, KIM T Y, SUMBT. Slot-Utterance Matching for Universal and Scalable Belief Tracking [C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019; 5478-5483.
- [34] FENG Y, WANG Y, LI H. A Sequence-to-Sequence Approach to Dialogue State Tracking [J]. arXiv:2011.09553, 2020.
- [35] QUAN J, XIONG D. Modeling Long Context for Task-Oriented Dialogue State Generation [C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, 2020; 7119-7124.
- [36] SANH V, DEBUT L, CHAUMOND J, et al. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter [J]. arXiv:1910.01108, 2019.



ZHU Ruo-chen, born in 1997, postgraduate, is a member of China Computer Federation. His main research interests include natural language processing and dialogue system.



ZHANG Deng-hui, born in 1970, master, professor, is a member of China Computer Federation. His main research interests include distributed service collaboration and natural language processing.